



Calhoun: The NPS Institutional Archive

Theses and Dissertations

Thesis Collection

1987

Calibration of a triple wire probe for turbulence measurements.

Selman, Gary J.

<http://hdl.handle.net/10945/22206>



Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>

11-11-60
 11-11-60
 11-11-60

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

541723

CALIBRATION OF A TRIPLE WIRE PROBE
FOR
TURBULENCE MEASUREMENTS

by

Gary J. Selman

December 1987

Thesis Advisor

J. Val Healey

Approved for public release; distribution is unlimited.

T239233

REPORT DOCUMENTATION PAGE

1. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS		
2. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release, distribution unlimited		
5. DECLASSIFICATION/DOWNGRADING SCHEDULE					
PERFORMING ORGANIZATION REPORT NUMBER(S)			5. MONITORING ORGANIZATION REPORT NUMBER(S)		
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School		6b. OFFICE SYMBOL (If applicable) CODE 67		7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School	
7. ADDRESS (City, State, and ZIP Code) Monterey CA, 93943-5100			7b. ADDRESS (City, State, and ZIP Code) Monterey CA, 93940-5100		
8. NAME OF FUNDING/SPONSORING ORGANIZATION		8b. OFFICE SYMBOL (If applicable)		9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
10. ADDRESS (City, State, and ZIP Code)			10. SOURCE OF FUNDING NUMBERS		
		PROGRAM ELEMENT NO.		PROJECT NO.	TASK NO.
				WORK UNIT ACCESSION NO.	
1. TITLE (Include Security Classification) Calibration of a Triple Wire Probe for Turbulence Measurements					
2. PERSONAL AUTHOR(S) Gary J. Selman, Ltcd, USN					
3a. TYPE OF REPORT Masters Thesis		13b. TIME COVERED FROM _____ TO _____		14. DATE OF REPORT (Year, Month, Day) 1987, December 1987	
15. PAGE COUNT 44					
6. SUPPLEMENTARY NOTATION					
7. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB-GROUP			
			Hot-Wire; Triple Wire Probe		
9. ABSTRACT (Continue on reverse if necessary and identify by block number)					
<p>A method of calibration for a triple wire probe has been implimented for use in the flow visualization tunnel at the Naval Postgraduate School. The system was designed to allow flow measurements in highly turbulent flows and under conditions of relatively unknown flow directions. Preliminary evaluation of the method indicates that accuracies in the range of 1-4% can be expected. Although attainment of these levels of accuracy have not yet been achieved, it is felt that they can be realized.</p>					
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED		
22a. NAME OF RESPONSIBLE INDIVIDUAL J. Val Healy			22b. TELEPHONE (Include Area Code) (408) 646-2491		22c. OFFICE SYMBOL 67 Ile

Approved for public release; distribution is unlimited.

Calibration of a Triple Wire Probe
for
Turbulence Measurements

by

Gary J. Selman
Lcdr, United States Navy
B.S., The Citadel, 1977

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1987

ABSTRACT

A method of calibration for a triple wire probe has been implemented for use in the flow visualization tunnel at the Naval Postgraduate School. The system was designed to allow flow measurements in highly turbulent flows and under conditions of relatively unknown flow directions. Preliminary evaluation of the method indicates that accuracies in the range of 1 - 4% can be expected. Although attainment of these levels of accuracy have not yet been achieved, it is felt that they can be realized.

541723
C.1

TABLE OF CONTENTS

I.	INTRODUCTION	7
I.	EQUIPMENT	9
	A. PROBE	9
	B. DATA COLLECTION EQUIPMENT	9
	C. CALIBRATION EQUIPMENT	10
II.	PROBE CALIBRATION	15
	A. GEOMETRIC DEFINITIONS	16
	B. CALIBRATION THEORY	16
	C. DATA REDUCTION	20
	D. MODIFICATIONS TO BASIC PROCEDURE	22
III.	SOFTWARE GUIDE	23
	A. CALIBRATION PROCEDURES	23
	B. DATA COLLECTION	25
	C. DATA REDUCTION	26
	D. GENERAL NOTES ON THE SOFTWARE	26
IV.	CONCLUSIONS	28
V.	RECOMMENDATIONS	29
	APPENDIX	31
	REFERENCES	42
	INITIAL DISTRIBUTION LIST.	43

LIST OF FIGURES

1. Velocity Calibration	12
2. Probe Positioner	13
3. Velocity Profile	14
4. Coordinate System	17
5. Velocity Magnitude Calibration	21
6. Wire Angle Calibration	21

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my loving wife, Lynda, without whose support the completion of this research would have been impossible. Her understanding of all those late nights and her comfort when things went wrong will always be remembered.

I would also like to thank my Thesis Advisor, Professor Val Healey, who provided me with the opportunity to do this research.

Last but not least, I thank Lcdr. Tom Cahill, who by his help in areas outside of this research, provided me the time to complete this project.

I. INTRODUCTION

A method of calibration of a three component constant temperature anemometer (CTA) and the reduction of data obtained from the system is discussed. This report is intended to document the system for future reference in the study of flow around ship models currently being conducted at the Naval Postgraduate School (NPS).

Recent work in the Flow Visualization Tunnel at NPS has successfully modeled the atmospheric boundary layer that exists in open ocean conditions [Ref. 1]. Extensive on- and off-body flow visualization studies have also been conducted to determine the flow field around a generic destroyer model. Work currently being completed will provide the characteristics of the flow around an oscillating model. These studies are being conducted as part of a study to determine the feasibility of simulating the ship/helicopter interface with sufficient realism to permit certification of helicopter operations on those ship decks not currently certified.

The next step in this study is to map the flow field around the model. The three components of mean velocity, flow directions, turbulence intensities, and spectrum functions are desired. The above-mentioned work has demonstrated the extremely turbulent nature of the flow as well as the locations of reverse flow.

Historically, analog circuits have been used to reduce wire voltages to velocity components. This has been due, in part, to the amount of computation time required per data point. Butler and Wagner [Ref. 2] and Westphal and Metha [Ref. 3] have derived methods of calibration and measurement that have achieved a considerable reduction in required computational and calibration time.

II. EQUIPMENT

A. PROBE

A commercial DISA triaxial probe, type 55P91 was used for measurements. The wires of the probe are arranged parallel to an orthogonal coordinate axis system that is rotated with respect to the probe axis system. The ends of the wires lie on the surface of a sphere that has a diameter of 3 mm, so that the sensitive volume of the probe is smaller than that of the 9 mm diameter of the head [Ref. 4, p. 17]. The reference position of the probe is with its number three wire in the X-Y plane with its shortest prong at the smaller value of Y.

B. DATA COLLECTION EQUIPMENT

A three channel Dantec constant temperature anemometer system was used for data collection. Each channel consisted of a 56C01 CTA with a 56C17 CTA Bridge Module. The output of the CTA was further processed by a 56N20 Signal conditioner that allowed filtering of unwanted frequency components. Also available in the system was a 56N25 Digital RMS

voltmeter and a 56N22 Digital Mean Value Unit. The latter two units were used to monitor performance of system software and hardware during flow measurement.

The outputs from the three CTA channels were connected to a 12 bit Metrabyte A/D converter where they were collected and stored for reduction. The A/D board uses a 12 bit successive approximation converter with a 12 microsecond conversion time yielding a maximum throughput rate of 60 KHz in D.M.A mode [Ref. 5, p. 1]. Analog inputs are sampled in sequence by means of a multiplexer which passes the data to a sample and hold circuit where they are held until completion of the previous conversion. This causes a delay of approximately one conversion time between successfully scanned channels. Due the relatively low frequency nature of the signals to be studied (below 1KHz), This delay was not considered significant.

C. CALIBRATION EQUIPMENT

Calibration was conducted in a small wind tunnel with a 5" X 5" interior cross section. Static pressure ports were located along one side of the tunnel to provide test section velocities. The static pressure was measured by means of a

micromanometer with one port connected to a static pressure port opposite the probe head. The other port was open to atmospheric pressure. Actual velocity was derived from a calibration curve relating the static pressure differential to velocities obtained from a pitot-static tube (Fig. 1). Due to the small size of the tunnel, it was undesirable to mount a pitot-static probe in the tunnel during calibration.

The probe positioner was designed to provide $\pm 45^\circ$ of pitch and $\pm 90^\circ$ of yaw. and was mounted on top of the calibration tunnel. Figure 2 shows a schematic of the positioner and probe holder. The pivot point for pitch and yaw were coincident and are located on the center line of the tunnel. Yaw was achieved by means of a stepper motor under the control of the calibration program. This stepper motor allowed for yaw increments of 15° per step. During calibration, pitch was adjusted manually. Markings of $\pm 5, 10, 20, 30, 40$, and 45° were etched on the positioner arch.

The stem of the probe holder was offset so that the head of the probe was approximately 1.5" ahead of the pivot point of the system. This arrangement allowed for maximum pitch and yaw ranges while minimizing probe head travel. A survey

of the velocity profile at the calibration point of the tunnel indicated that the velocity was in fact constant over the range of travel (Fig 3).

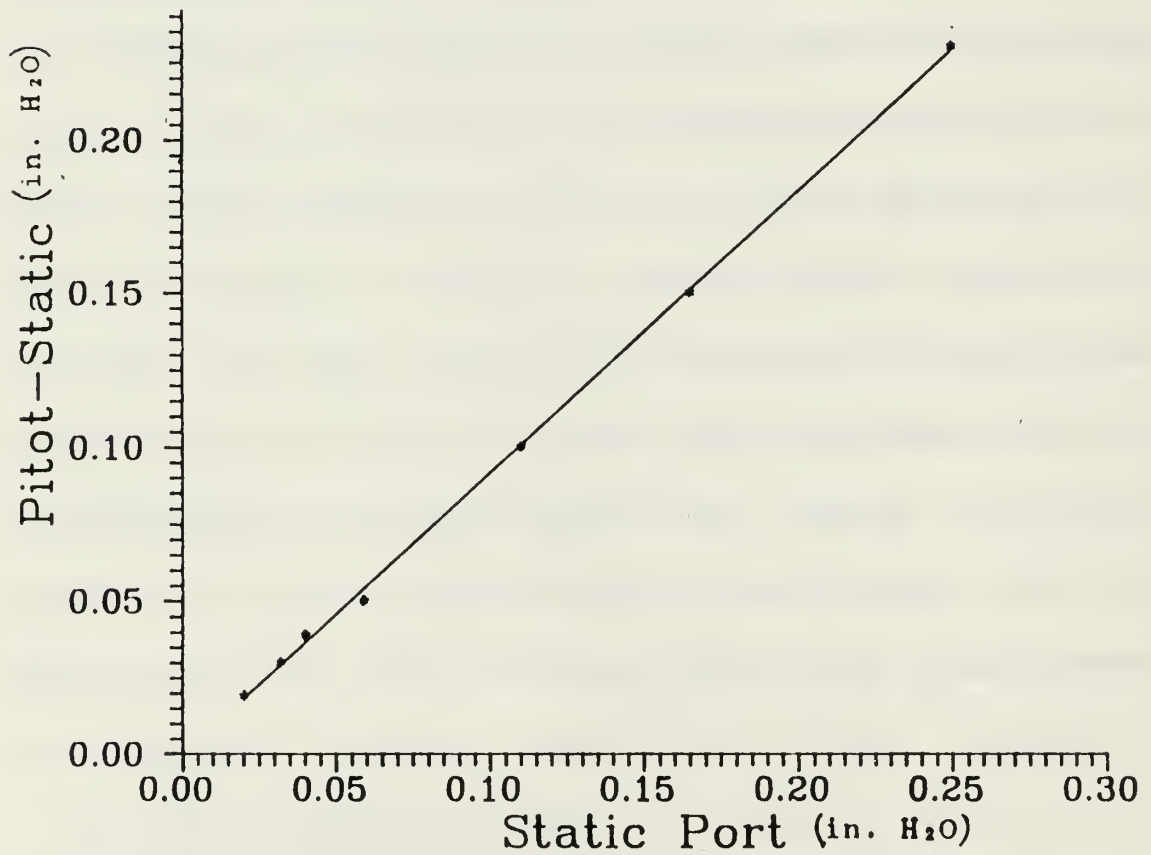


Figure 1. Velocity Calibration

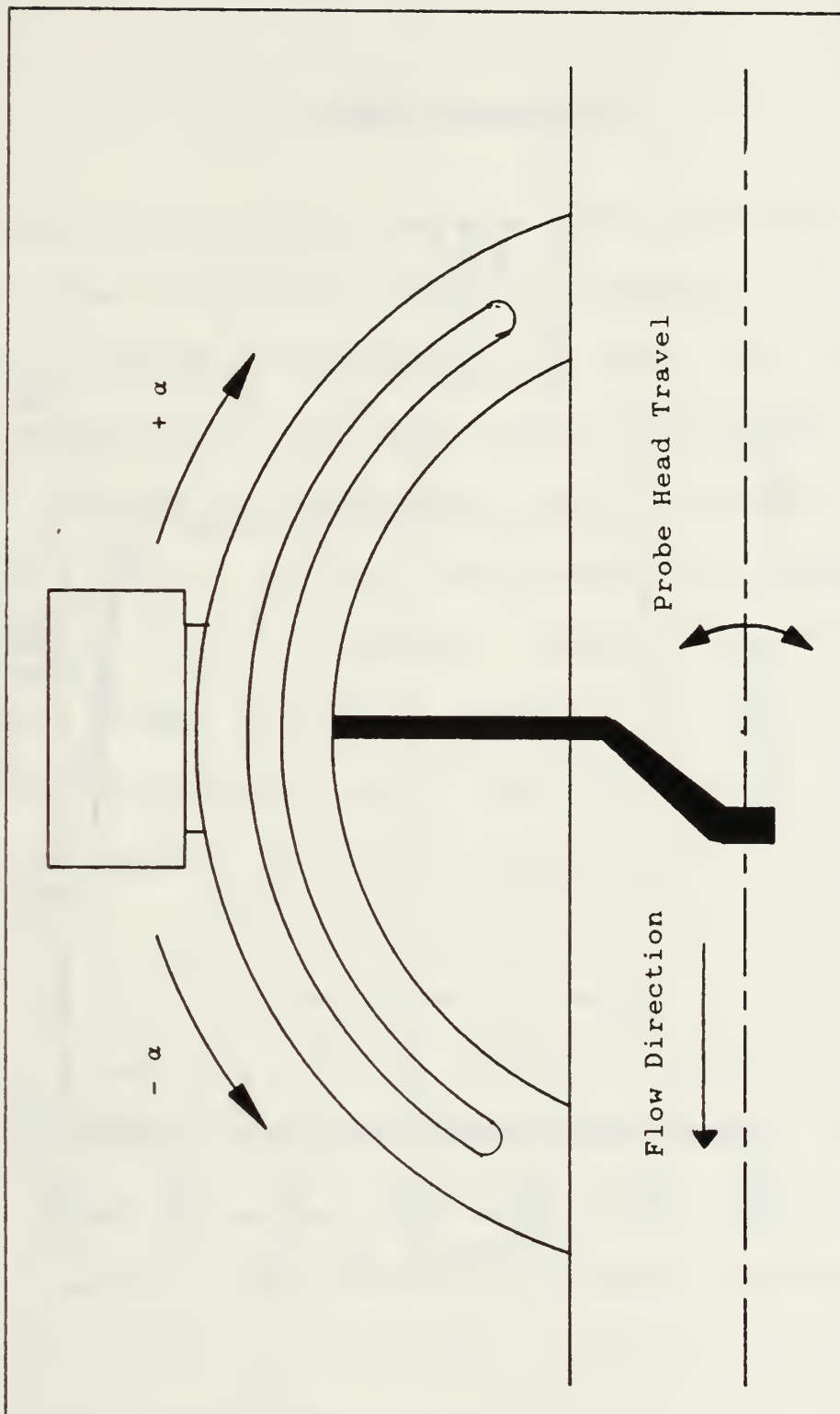


Figure 2. Probe Positioner

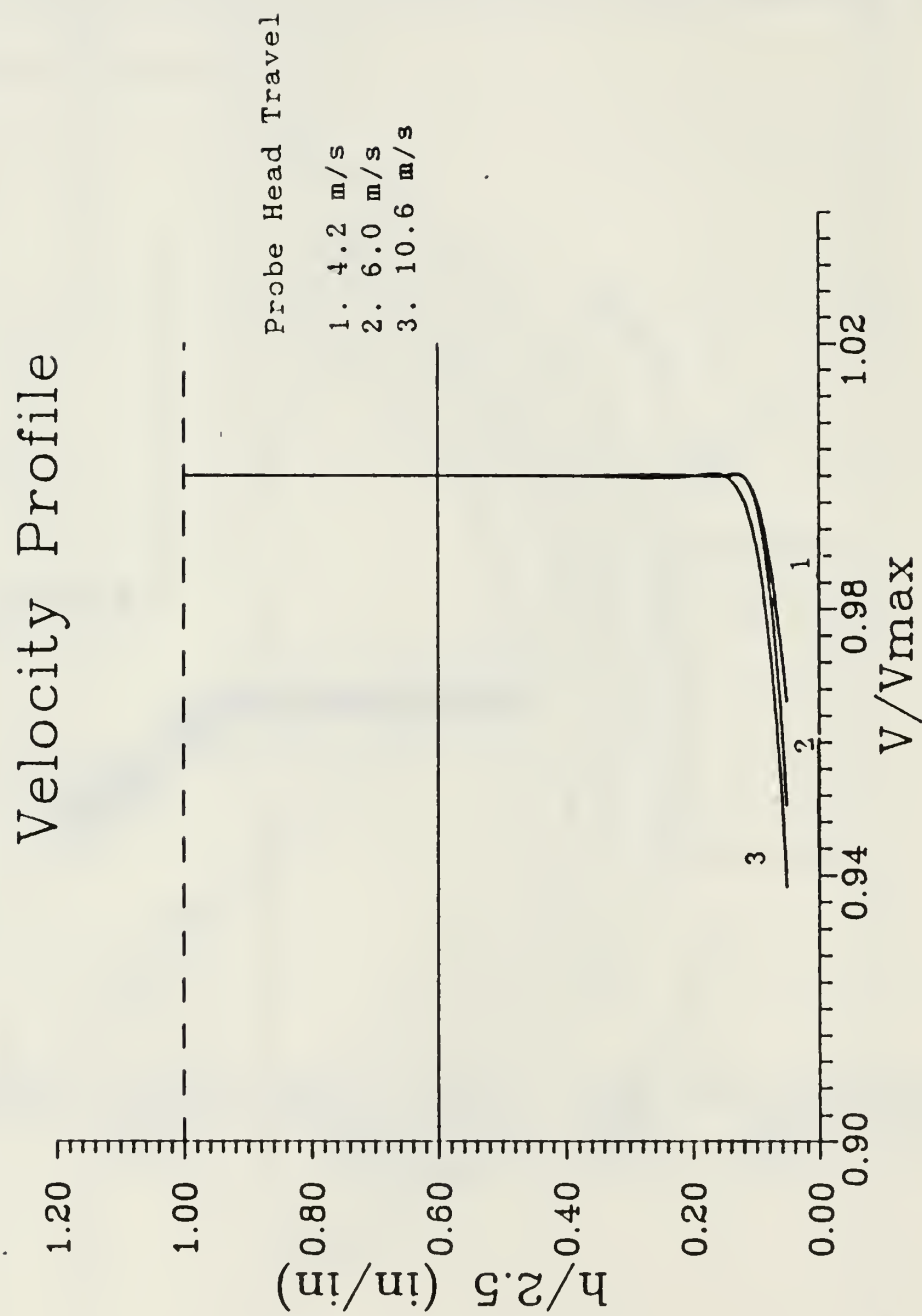


Figure 3. Velocity Profile

III. PROBE CALIBRATION

As previously mentioned, a commercially available probe having three independent, mutually orthogonal wires was utilized. A method of calibration was sought that provided rapid reduction of data and reduced calibration time. Several methods of calibration were investigated that provided highly reliable measurements of mean and fluctuating velocity components. However, in flows of relatively large turbulence intensity and unknown flow directions, these methods tend to produce large errors [Ref. 2, p. 1]. As pointed out in the introduction, the calibration process had to provide accurate results when subjected to this type of flow condition. A search of the current literature indicated that the method of Butler and Wagner [Ref. 2] was well suited to the problem at hand. Not only did it extend the yaw and pitch envelope but it also provided for a relatively rapid calibration process. This method is discussed in detail in Reference 1 and an overview with slight modifications is given here.

A. GEOMETRIC DEFINITIONS

Wire, probe, and Laboratory coordinate systems are defined in Figure 4. Pitch (α) is defined as the angle between projection of the velocity vector onto the X-Z plane and the X-axis. Yaw (β) is the angle between the projection of the vector in the X-Y plane and the X axis. The positive sense of these two angles is as shown in the diagram. The angle θ_i is the angle between the velocity vector and wire i where $i = 1, 2, 3$.

B. CALIBRATION THEORY

The underlying theory of the calibration procedure is that if the angle between any two wires of the probe and the velocity vector is known, the pitch and yaw of the velocity vector in laboratory coordinates can be derived. Using this method, there are two possible directions of the velocity vector. The angle between the third wire and the velocity vector is used to resolve the proper direction.

If the components of a vector are known in the wire coordinates, then the components of that vector in the laboratory coordinate system can be found by the following equation:

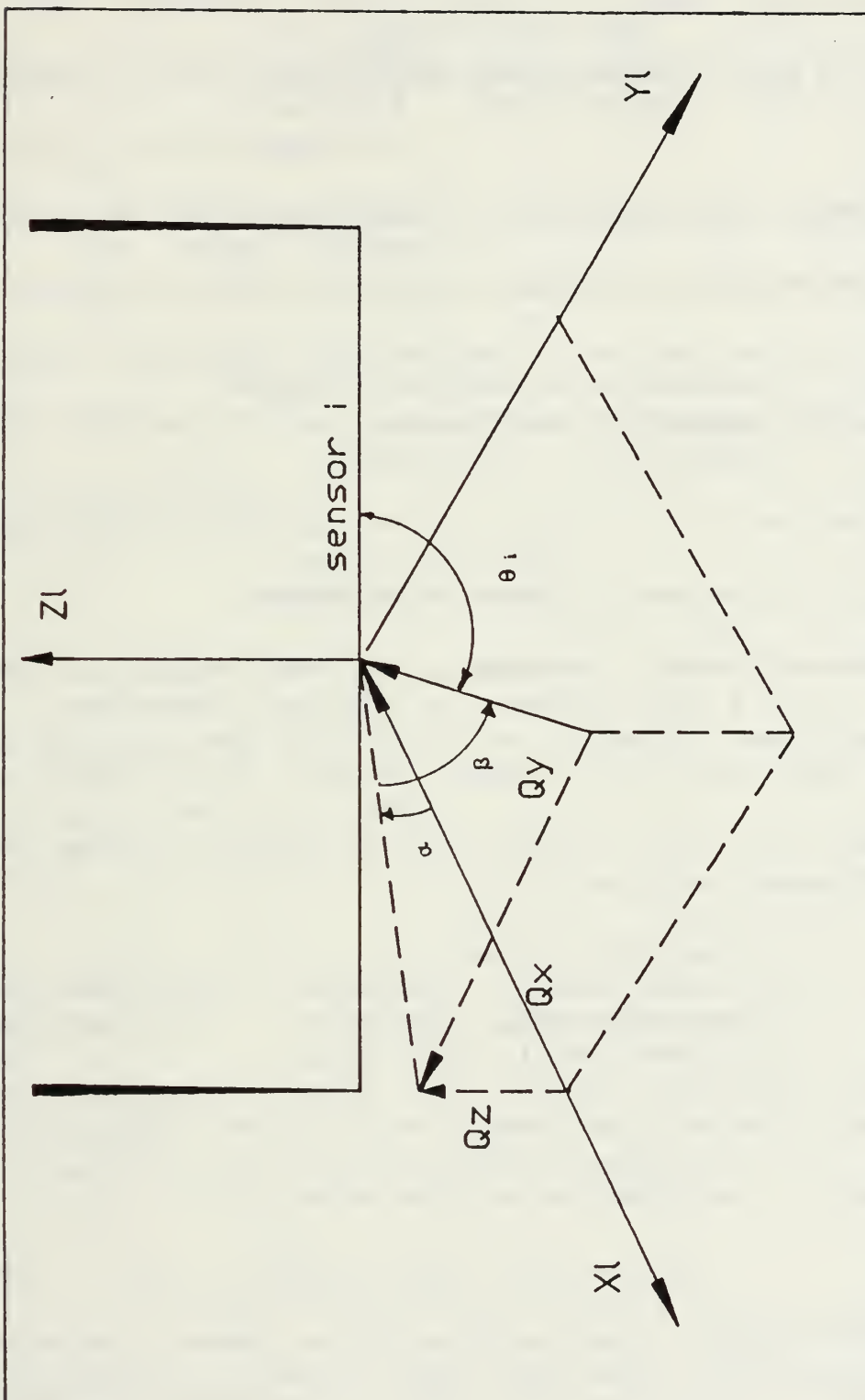


Figure 4. Coordinate System

$$\begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix} = \begin{bmatrix} A \end{bmatrix} \begin{bmatrix} \sqrt{3}/3 & \sqrt{3}/3 & \sqrt{3}/3 \\ \sqrt{6}/6 & \sqrt{6}/6 & -\sqrt{6}/3 \\ \sqrt{2}/2 & -\sqrt{2}/2 & 0 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix}$$

$$A \equiv \begin{bmatrix} \cos(\alpha) & 0 & -\sin(\alpha) \\ 0 & 1 & 0 \\ \sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \begin{bmatrix} \cos(\beta) & \sin(\beta) & 0 \\ -\sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$X_L, Y_L, Z_L \equiv$ Components in laboratory system

$\alpha, \beta \equiv$ pitch and yaw of probe axis

Eqn. 1

$X_w, Y_w, Z_w \equiv$ Components in wire system

Likewise, the velocity vector can also be described in terms of the laboratory coordinate system:

$$Q = -Q(\cos\alpha)(\cos\beta)i - Q(\sin\beta)j - Q(\sin\alpha)(\cos\beta)k \quad \text{Eqn. 2}$$

$i, j, k \equiv$ unit vectors ; $Q \equiv$ Velocity Magnitude

An expression for the angle between the velocity vector and any wire unit vector can be found by taking the dot product of the two:

$$\begin{aligned} \cos \theta_1 &= -\sqrt{3}/3(\cos\alpha)(\cos\beta) - \sqrt{6}/6(\sin\beta) - \sqrt{2}/2(\sin\alpha)(\cos\beta) \\ \cos \theta_2 &= -\sqrt{3}/3(\cos\alpha)(\cos\beta) - \sqrt{6}/6(\sin\beta) + \sqrt{2}/2(\sin\alpha)(\cos\beta) \\ \cos \theta_3 &= -\sqrt{3}/3(\cos\alpha)(\cos\beta) + \sqrt{6}/3(\sin\alpha) \end{aligned} \quad \text{Eqn. 3}$$

The following relationship was used to relate the wire voltages to the effective cooling velocity:

$$Q_{effi} = K_1 + K_2 E_i + K_3 E_i^2 \quad \text{Eqn. 4}$$

The effective cooling velocity was taken to be equal to the total velocity when the wire was oriented normal to the calibration flow. The coefficients of Equation 3 for each

wire were determined by varying the tunnel velocity with each sensor in this orientation and then performing a least squares fit on the data.

Probe angular response was calibrated by varying the pitch and yaw of the probe with respect to the centerline of the tunnel. Yaw and pitch angles as well as the three wire several different tunnel velocities. The range of yaw and pitch were as follows:

$$\begin{aligned} -30 &\leq \beta \leq 30 \text{ deg} \\ -40 &\leq \alpha \leq 40 \text{ deg} \end{aligned}$$

The following correlating parameters were then used to obtain accurate data fits for the velocity magnitude and sensor angles [Ref. 2, p. 728] :

$$X \equiv \sqrt{(E_1^2 + E_2^2 + E_3^2)} \quad \text{Velocity Magnitude Parameter}$$

$$Y \equiv E_1 / \sqrt{(E_2^2 + E_3^2)} \quad \text{Pitch Angle Parameter}$$

$$Z \equiv E_2 / E_3 \quad \text{Yaw Angle Parameter}$$

The Calibration data for velocity magnitude were then least squares fitted to the following equation:

$$\begin{aligned} Q = & A_0 + A_1X + A_2X^2 + A_3Y + A_4Y^2 \\ & + A_5Z + A_6Z^2 + A_7XY + A_8XZ + A_9YZ \end{aligned} \quad \text{Eqn. 5}$$

Calibration for sensor angle were then fit according to the following :

$$\begin{aligned}
\theta_i = & B_0 + B_1Q + B_2Q^2 + B_3F_i + B_4F_i^2 \\
& + B_5Y + B_6Y^2 + B_7Z + B_8Z^2 + B_9QF_i \\
& + B_{10}QY + B_{11}QZ + B_{12}F_iY + B_{13}F_iZ + B_{14}YZ
\end{aligned}
\tag{Eqn. 6}$$

The least squares curve fitting of the above two equations was accomplished utilizing MATLAB, a commercially available program by Mathworks. The use of this program greatly simplified the programming effort as well as providing extremely rapid data reduction. Calibration data for one wire is shown in Figures 5 and 6.

C. DATA REDUCTION

Once the simultaneous sensor output voltages have been collected from the wires, the three parameters given in Section B are computed and substituted into Equation 5. The velocity magnitude is then utilized in Equation 6 to obtain the wire angle for each of the three wires.

As previously noted, there are two possible solutions for the pitch and yaw angle of the velocity vector given any two of the wire angles. The details of the procedure to resolve the proper solution is given in the Appendix of Reference 2.

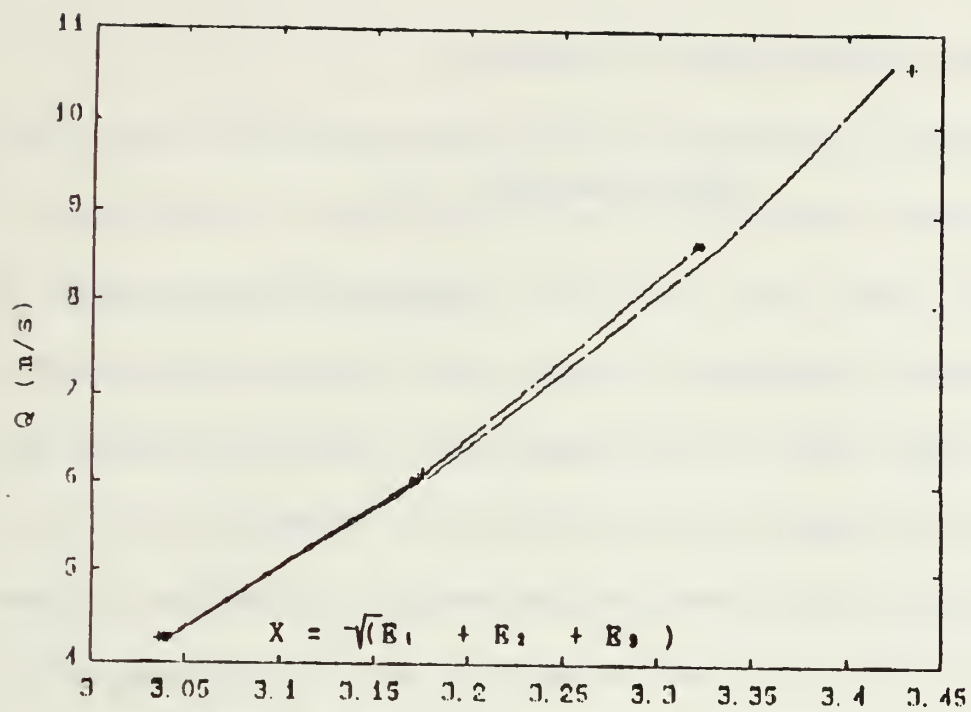


Figure 5. Velocity Magnitude Calibration

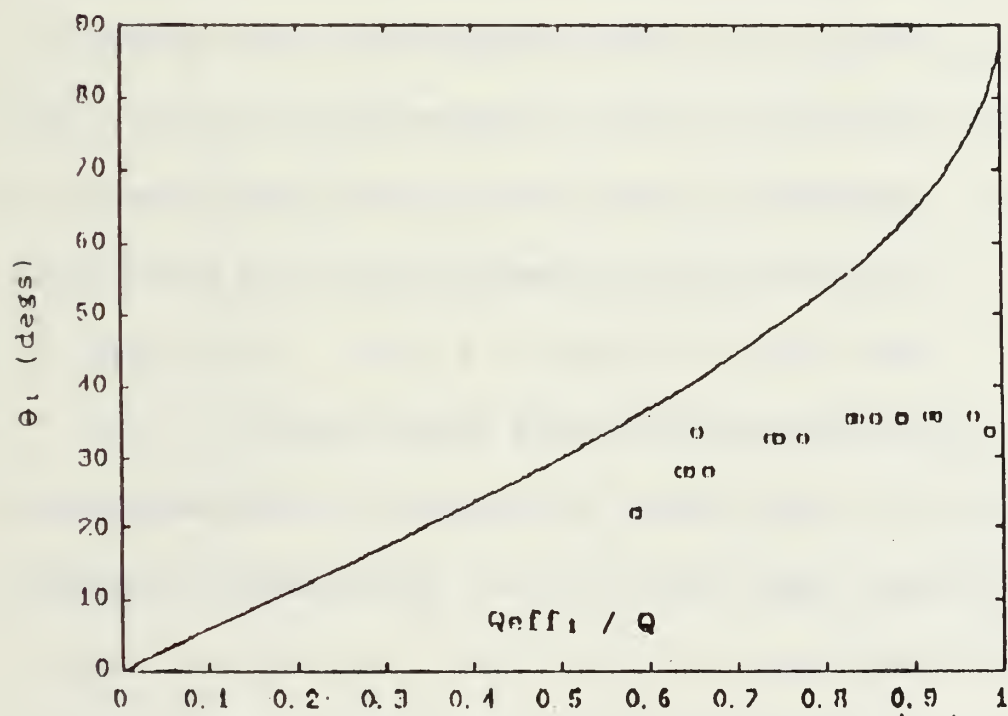


Figure 6. Wire Angle Calibration

D. MODIFICATIONS TO BASIC PROCEDURE

It should be noted that the definitions of a and b have been reversed from those given in Reference 2. The origin of the pitch angle has also been modified. This was done to bring these definitions more into line with current aeronautical usage. The appropriate changes to Reference 2 equations were made to account for this change.

Reference 6 discusses two methods by which the accuracy of hot wire data obtained using this particular probe may be improved:

- (1) Adjustment of the overheat control of two of the wires to match the calibration of the third.
- (2) Compensation for temperature drift away from calibration conditions during measurement by adjustment of overheat controls to produce the same output voltages at a given tunnel velocity as was obtained during calibration.

Method 1 was used during testing. Due to the temperature stability of the laboratory facility, the second method has not been utilized.

IV. SOFTWARE GUIDE

A. CALIBRATION PROCEDURES

Probe calibration consists of two separate procedures, the calibration of each individual wire and velocity magnitude/yaw-pitch calibration. The following steps will ensure that the necessary procedures will be carried out in the proper order.

Step 1.

Ensure that no memory resident programs are currently running as this will interfere with the proper operation of some of the data fitting routines. Start the calibration routine by typing "CALIBRAT" at the DOS prompt. This will provide the user with a menu of choices.

Step 2.

Each individual wire must first be calibrated to obtain the coefficients of Equation 4. Selection of option 1, "King's Law Calibrate", will start this portion of the calibration process. Each wire in turn must be placed perpendicular to the calibration flow during this procedure. This involves a rotation of the probe about its X-axis and a

yaw about the Z-axis. The X-axis rotation is done manually and is accomplished by loosening the set screw that holds the probe in the probe holder. Proper positioning is accomplished by locating one of the three pattern of dots that indicate the individual wires on top of the probe axis. The software calibration routine provides on screen documentation of the procedure. After positioning the probe according to the directions, the software yaws the probe so as to bring the wire under calibration perpendicular to the flow. After the calibration data has been collected, the procedure is repeated until all wires have been calibrated.

The software will generate a file called "VELCAL.M" which contains the data for each wire. Data is stored with the voltage in Column 1 and the calibration velocity in Column 2. This data is then fit by a MATLAB file called "DATAFIT". The results are stored in a file titled "KPARAM" which is later used by the data reduction program as well as the MATLAB file that performs the curve fitting on the velocity and wire angle data. Entry into the matlab routines is automatic and no user interaction is required.

Step 3.

Data for velocity magnitude and wire angle calibration is collected simultaneously. A minimum of 15 data points at various yaw, pitch and velocity settings are required to complete the process. After selection of option 2, "Yaw/Pitch Calibrate", answer the questions as they are asked. Probe pitch angle must be set manually while the selected yaw is set by the stepper motor. Be sure to check that the tunnel velocity is set to the desired speed prior to collecting the data. The latter is stored by the program in two files, YPCAL1.M and YPCAL2.M, which are later joined by MATLAB for use in curve fitting.

B. DATA COLLECTION

The data collection routine asks for the probe position in X,Y,Z format and stores the response as the first entry in a file named "DATA1". Enter the number of samples desired to be collected at the data point, in response to the prompt. The program then places the specified number of recorded voltages in the same data file as the point coordinates.

The program assumes that the Probe is aligned with the flow and therefore does not request yaw and pitch settings. if another orientation of the probe is desired, it must be recorded at the time of testing. Yaw is set by selecting Option 5, "Yaw Probe", from the main menu.

C. DATA REDUCTION

Data reduction is accomplished by selecting Option 4 from the CALIBRATE opening menu or by typing "REDUCE" at the DOS prompt. The program will ask for the name of the data file to reduce and for the name of file in which to store the results. Program operation proceeds automatically and upon completion of the reduction, the program asks if the user wishes to perform another reduction. Data from this routine is stored with velocity magnitude in column 1, pitch angle in column 2 and yaw angle in column 3.

D. GENERAL NOTES ON THE SOFTWARE

Raw voltage data was collected from the probe using a compiled basic program driver for a machine language subroutine supplied with the A/D Board. The source listing of the Basic Language program can be found in the appendix. During this preliminary concept exploration phase, the DMA

capabilities of the hardware and software were not utilized due to problems encountered with the software supplied with the A/D board. Specifically, when Microsoft Quick Basic dynamically dimensions arrays, it places them outside of the current memory segment. The DASH16 machine language driver supplied with the A/D board expects to find the arrays in the current segment. When this condition occurs, the system crashes. It should be noted that data collection rates of up to 60 KHz may be achieved by utilizing these capabilities.

As previously mentioned, a commercially available program, Matlab, was used to perform the least squares curve fitting of the calibration data. Matlab is a matrix manipulation program and provided very rapid reduction of large amounts of data.

V. CONCLUSIONS

The selected method of calibration promises to provide accurate and rapid reduction of data. System software has been completed and integrated with the calibration hardware. Preliminary results show excellent velocity accuracy. Sensor angle calibration still displays considerable errors over the range tested. This is currently thought to be due to errors in the calibration routine vice the calibration method. It is also felt that the accuracy of the data fit will improve after an initial learning period, during which time the best selection of yaw and pitch angles for calibration can be determined.

VI. RECOMMENDATIONS

A. EQUIPMENT

The current configuration of the probe positioner / holder causes considerable difficulty in aligning the probe with the center line of the tunnel. This could be rectified by cutting the probe holder shaft below the sliding block on the positioner and coupling it together with a sleeve and set screw arrangement. This would allow for rotation of the probe independent of the stepper motor.

The present calibration system requires manual settings of multiple pitch angles and constant monitoring of the tunnel velocity. The micromanometer used to determine tunnel velocity also requires constant attention to insure drift from the zero reference point has not occurred. This combination of shortcomings increases the required calibration time. Considering the frequency with which recalibration is required to maintain accuracy, replacement of the present system with a DISA Calibrator Unit would reduce the tedium of the process as well as improve calibration results.

B. PROBE

Probe geometry has been assumed to be perfectly orthogonal during the development of the algorithms. Improved accuracies may well be obtained by direct measurement of the angles of the wires. It has recently come to the attention of the author that an optical comparator is available at NPS to perform these measurements.

Reference 6 provides an indepth investigation of the effects of roll angles on the accuracy of data obtained from the particular probe being used. This information should be utilized when actual measurements are performed.

APPENDIX

```

'*****
'*
'*          Probe Calibration Routine
'* Gary J. Selman          Ver. 1.0
'* Naval Postgraduate School      14 November 1987
'*****
    DIM DIO%(4)
    COMMON SHARED DIO%(),L%,U%,TEMP      'Make available to all routines

'*****
'*
'*          START OF FUNCTION DEFINITIONS
'*
'*
'*****

DEF FNVelocity(Delta.P)
    STATIC delp
    delp=-.00202 + .93*Delta.P          'Converts delta p from static
                                         'port to equivalent pitot static
                                         'delta p
    FNVelocity =(2.9 * SQR((TEMP+459.0)*delp))* .3048
END DEF

DEF FNDegRad(DEG)=DEG * (Pi/180)          'Degrees To Radians
DEF FNVelocityParam(E1,E2,E3)=SQR(E1^2+E2^2+E3^2) 'Velocity Parameter
DEF FNPitchParam(E1,E2,E3)=E1/(SQR(E2^2+E3^2)) 'Pitch Parameter
DEF FNYawParam(E2,E3)=E2/E3              'Yaw Parameter

'*****
'*          SUBROUTINE SAMPLE
'* Samples each wire specified L% and U%. Note that these two values
'* have been declared common and are available to all routines
'* ARRAY : Three dimension Array that returns the mean
'*          Voltage of each wire sampled
'* NoSamples : Specifies how many samples will be used to calculate
'*          the mean Voltage. If = 1, returns only an instantaneous
'*          voltage
'* FirstDim : Specifies first dimension of ARRAY.
'* DIO%(0) : Returned by DASH16 = Voltage read by A/D
'* DIO%(1) : Returned by DASH16 = A/D Channel read
'* L% & U% : Lower and upper channel scan limits

SUB Sample(NoSamples%,FirstDim,ARRAY(3)) STATIC      'Note that L% & U%
                                                    'are shared

DIM SumVolt#(3)
    ERASE SumVolt#

```

```

    STATIC MeanVoltage#,I,J
    MeanVoltage#=0.0
    MD% = 3
    FOR I=1 TO NoSamples%
        FOR J = L% TO U%
            CALL DASH16 (MD%, DIO%(0), FLAG%)
            IF FLAG%<>0 THEN PRINT "Error #";FLAG%;" in mode"
            SumVolt#(DIO%(1))=((5.*DIO%(0))/2047)+SumVolt#(DIO%(1))
            MeanVoltage#=SumVolt#(DIO%(1))/I
            ARRAY(FirstDim,1,DIO%(1)) = MeanVoltage#
        NEXT J
    NEXT I
END SUB

'*****
SUB INITIALIZE.AD STATIC 'initial with mode 0
    OPEN "DASH16.ADR" FOR INPUT AS #1 'get base I/O address
    INPUT #1, DIO%(0)
    CLOSE #1
    DIO%(1) = 2 'interrupt level
    DIO%(2) = 3 'D.M.A. level
    DASH16 = 0 'call offset - always zero
    FLAG% = 0 'error variable
    MD% = 0 'mode 0 - initialize
    CALL DASH16 (MD%, DIO%(0), FLAG%)
    IF FLAG%<>0 THEN PRINT "INSTALLATION ERROR":STOP 'Halt on error
END SUB

'*****
SUB GET.SCAN.LIMITS(L%,U%) STATIC ' set scan limits
    'Scan limits will default to 0-7 (8 channel) or 0-15 (16 channel) if
    you
    'skip this step
    INPUT "Enter lower channel scan limit number : ";L%
    INPUT "Enter upper channel scan limit number : ";U%
END SUB

'*****
SUB SET.SCAN.LIMITS(L%,U%) STATIC
    DIO%(0)=L% 'set lower limit
    DIO%(1)=U% 'set upper limit
    MD% = 1 'mode 1 - set scan limits
    CALL DASH16 (MD%, DIO%(0), FLAG%)
    IF FLAG%<>0 THEN PRINT "Error #";FLAG%;" in setting scan limits":STOP
END SUB

'*****
SUB MATLAB.OUT(ARRAY(3)) STATIC
    STATIC I,J,K
    DIM Wire$(3)
    Wire$(0)="NoWires="
    Wire$(1)="wire1=["
    Wire$(2)="wire2=["
    Wire$(3)="wire3=["

```



```

PRINT " The program will generate a file called VELCAL.M"
PRINT " This file will contain the data collected for each"
PRINT " wire in a format suitable for execution by MATLAB."
PRINT " MATLAB will perform a least squares curve fit on the"
PRINT " data by utilizing a program called DataFit.M"

```

```

NAME$="C:\PROBCAL\VELCAL.M"
OPEN NAME$ FOR OUTPUT AS #2
PRINT#2,Wire$(0),UBOUND(ARRAY,3)
FOR I=1 TO UBOUND(ARRAY,3)
IF I <>1 THEN PRINT#2 ,"J"
PRINT#2 ,Wire$(I),
FOR J=1 TO UBOUND(ARRAY,1)
    FOR K = 1 TO UBOUND(ARRAY,2)
        PRINT#2,ARRAY(J,K,I),
    NEXT K
    PRINT#2,
NEXT J
NEXT I
PRINT#2,"J"
CLOSE#2

```

```

END SUB

```

```

'*****

```

```

SUB Velocity.Sensitivity(No.Of.Wires) STATIC

```

```

    INPUT " How many calibration points will you enter";Points

```

```

    REDIM CAL.ARRAY (Points,2,No.Of.Wires)

```

```

    FOR I = 1 TO No.Of.Wires

```

```

        L% = I : U% = I

```

```

        CALL SET.SCAN.LIMITS(L%,U%)

```

```

        CLS

```

```

        PRINT USING " Set Wire # perpendicular to flow";I

```

```

        IF I = 1 THEN

```

```

            PRINT "CENTER THE 3 DOTS ON TOP OF THE PROBE"

```

```

            PRINT "PRESS ANY KEY WHEN READY:"

```

```

            PRINT "***** WARNING *****"

```

```

            PRINT "PROBE WILL ROTATE WHEN YOU DO"

```

```

HEREA:  A$ = INKEY$ : IF A$ = "" THEN GOTO HEREA

```

```

        Yaw = -45

```

```

        CALL YawProbe(1,Yaw)

```

```

    END IF

```

```

    IF I = 2 THEN

```

```

        PRINT "CENTER THE SINGLE DOT ON TOP OF THE PROBE"

```

```

        PRINT "PRESS ANY KEY WHEN READY:"

```

```

        PRINT "***** WARNING *****"

```

```

        PRINT "PROBE WILL ROTATE WHEN YOU DO"

```

```

HEREB:  A$ = INKEY$ : IF A$ = "" THEN GOTO HEREB

```

```

        Yaw = -45

```

```

        CALL YawProbe(1,Yaw)

```

```

    END IF

```

```

    IF I = 3 THEN

```

```

        PRINT "CENTER THE 2 DOTS ON TOP OF THE PROBE"

```

```

        PRINT "PRESS ANY KEY WHEN READY:"

```

```

        PRINT "PRESS ANY KEY WHEN READY:"
        PRINT "**** WARNING ****"
        PRINT "PROBE WILL ROTATE WHEN YOU DO"
HEREC:   A$ = INKEY$ : IF A$ = "" THEN GOTO HEREC
        Yaw = -45
        CALL YawProbe(1,Yaw)
    END IF

    CLS
    PRINT USING "          Wire # Calibration Data ";I
    PRINT "          VOLTAGE          VELOCITY"
    FOR COUNT = 1 TO Points
        INPUT "Delta P from Micromanometer",DELP
        CALL Sample(1000,COUNT,CAL.ARRAY())
        CAL.ARRAY(COUNT,2,I)=FNVelocity(DELP)
        PRINT USING "          ###.####          ###.####";_
            CAL.ARRAY(COUNT,1,I),CAL.ARRAY(COUNT,2,I)
    NEXT COUNT
    Yaw = 0
    CALL YawProbe(1,Yaw)
NEXT I
    CALL MATLAB.OUT(CAL.ARRAY())
END SUB

SUB YawPitchParam(DELP,ALPHA,BETA) STATIC
CORRECT:
    INPUT " ENTER PROBE AXIS PITCH (DEGs) : ",ALPHA
    INPUT " ENTER PROBE AXIS YAW (DEGs) : ",BETA
    CALL YawProbe(1,BETA)
    INPUT " TUNNEL DELTA P : ",DELP
    INPUT " ANY CHANGES (Y/N) :";A$
    IF A$ = "Y" OR A$ = "y" THEN GOTO CORRECT
END SUB

SUB YawProbe(Mode%,Yaw) STATIC
    DIM Pointer(7)
    STATIC N,Position
    IF Mode% = 0 THEN
        CLS
        N = 3
        DATA 9,5,6,10,9,5,6,10
        FOR I = 0 TO 7
            READ Pointer(I)
        NEXT I
        PRINT " Insure probe is aligned with tunnel center line"
        print " "
        PRINT " DO NO CONNECT POWER TO STEPPER MOTOR
        PRINT " "
        PRINT " Press any key to continue"
HERE1:   A$ =INKEY$: IF A$ = "" GOTO HERE1
        MD% = 13
        DIO%(0) = Pointer(N)

```



```

        PRINT USING "DIO%(0) = ##"; DIO%(0)
        CALL DASH16(MD%,DIO%(0),FLAG%)
        IF FLAG% <> 0 THEN PRINT "Error ":STOP
        Position = 0
        'CLS
        PRINT "A valid code now exists on the"
print "input. Power may now be connected"
print
print "PRESS ANY KEY TO CONTINUE"
HERE2:      A$ = INKEY$ : IF A$ = "" GOTO HERE2

ELSE
    MD% = 13
    Direction = 0
    WHILE (Yaw <> Position)
        PRINT USING "Yaw = ##.##  Position = ###.##";Yaw,Position

        IF Yaw > Position THEN Direction = -1 ELSE Direction = 1
        N = N + Direction
        IF N < 0 AND Direction = -1 THEN N = 7
        IF N > 7 AND Direction = 1 THEN N = 0
        DIO%(0) = Pointer(N)
        CALL DASH16(MD%,DIO%(0),FLAG%)
        FOR Delay = 1 TO 5000
        NEXT Delay
        Position =Position - Direction*15
    WEND

    PRINT USING "Yaw = ##.##  Position = ###.##";Yaw,Position

END IF
END SUB

'*****
'*
'*          START OF MAIN PROGRAM
'*
'*****
    CALL INITIALIZE.AD          'Initialize A/D Converter
    CALL YawProbe(0,1)
    SCREEN 8,0,1,1:WIDTH 80:CLS:KEY OFF
    Pi = 3.14159263
MAIN:
    CLS
    PRINT "Do you want to      1.) King's Law Calibrate "
    PRINT "                    2.) Yaw / Pitch calibrate "
    PRINT "                    3.) Collect Data "
    PRINT "                    4.) Reduce Data"
    PRINT "                    5.) Yaw Probe
    PRINT "                    6.) Quit "
    INPUT "Enter Your Choice";A
    ON A GOTO KING,YAW.PITCH,COLLECT,REDUCE,SetYaw,QUIT

```

```

        INPUT "Enter Ambient Temperature";TEMP
        INPUT "How Many Wires ";No.Wires
        CALL Velocity.Sensitivity(No.Wires)
        GOTO MAIN

YAW.PITCH:
    SCREEN 8,0,2,1
    PRINT "                PRESS ANY KEY TO CONTINUE"
    SCREEN 8,0,1,1
    CLS
    DIM E(1,1,3)
    OPEN "C:\PROBCAL\YPCAL1.M" FOR OUTPUT AS #3
    OPEN "C:\PROBCAL\YPCAL2.M" FOR OUTPUT AS #4
    PRINT#3,"yp1=[",
    PRINT#4,"yp2=[",
    PRINT "This routine assumes that 3-D probe calibration is in
progress"
    INPUT " AMBIENT TEMPERATURE : " ,TEMP
    CALL GET.SCAN.LIMITS(L%,U%)
    CALL SET.SCAN.LIMITS(L%,U%)

AGAIN:
    CALL YawPitchParam(DELP,ALPHA,BETA)
    count=1.
    CALL Sample(1000,count,E())
    E1=E(1,1,1)
    E2=E(1,1,2)
    E3=E(1,1,3)
    CalVelocity=FNVelocity(DELP)
    X=FNVelocityParam(E1,E2,E3)
    Y=FNPitchParam(E1,E2,E3)
    Z=FNYawParam(E2,E3)
    CalVelocity=FNVelocity(DELP)
    SCREEN 8,0,2,2
    PRINT " Velocity      Yaw      Pitch      Sensor Voltage"
    PRINT USING "    ##.###      ##.###      ##.###
##.###";CalVelocity,_
                BETA,ALPHA,E1
    PRINT USING "                                ##.###";E2
    PRINT USING "                                ##.###";E3
    HERE:  A$=INKEY$: IF A$="" GOTO HERE
    SCREEN 8,0,1,1

    ALPHA=FNDegRad(ALPHA)
    BETA=FNDegRad(BETA)
    PRINT#3,CalVelocity,ALPHA,BETA,X,Y
    PRINT#4,Z,E1,E2,E3
    INPUT "DO YOU WANT TO ENTER CALIBRATION ANOTHER POINT";A$
    IF A$ = "Y" OR A$ = "y" GOTO AGAIN
    PRINT#3,"]"
    PRINT#4,"]"
    CLOSE #3
    GOTO MAIN

```

```

COLLECT:
    CLS

    CALL GET.SCAN.LIMITS(L%,U%)
    CALL SET.SCAN.LIMITS(L%,U%)
    DataPoint = 1
CollectAgain:

    INPUT " Enter X,Y,Z coordinates of sample",X,Y,Z
    INPUT " Enter Number Of Samples : ",COUNT
    FILE$ = "C:\PROBCAL\DATA"+STR$(DataPoint)
    OPEN FILE$ FOR OUTPUT AS #2
    REDIM SDATA(COUNT,1,3)
    REDIM S1DATA(COUNT,3)
    S1DATA(0,1)=X:S1DATA(0,2)=Y:S1DATA(0,3)=Z
    PRINT #2 ,X,Y,Z
    FOR I = 1 TO COUNT
        CALL Sample(1,I,SDATA())
    NEXT I
    FOR I=1 TO COUNT
        FOR J = 1 TO 3
            S1DATA(I,J)=SDATA(I,1,J)
            PRINT#2,S1DATA(I,J),
        NEXT J
        PRINT#2,
    NEXT I
    CLOSE #2
    DataPoint = DataPoint + 1

    PRINT "Do you want to : 1) Reduce this data now"
    PRINT "                        2) Sample another position"
    PRINT "                        3) Quit"
    INPUT "Enter your choice : ";A
    ON A GOTO REDUCE,CollectAgain,MAIN
REDUCE:
    RUN "REDUCE"
SetYaw:
    INPUT "Enter desired yaw angle : ",Yaw
    CALL YawProbe(1,Yaw)
    GOTO MAIN
QUIT:
END

```

```

'*****
'*
'*          3-Wire Probe Data Reduction Routine          *
'* Gary J. Selman                      Ver. 1.0          *
'* Naval Postgraduate School                14 November 1987 *
'*****

'*****
'*
'*          START OF FUNCTION DEFINITIONS                *
'*
'*****
COMMON SHARED Pi
DEF FNVelocity(Delta.P)
    STATIC delp
    delp=-.00202 + .93*Delta.P      'Converts delta p from static
                                   'port to equivalent pitot static
                                   'delta p
    FNVelocity =(2.9 * SQR((TEMP+459.0)*delp))* .3048
END DEF

DEF FNDegRad(DEG)=DEG * (Pi/180)      'Degrees To Radians
DEF FNVelocityParam(E1,E2,E3)=SQR(E1^2+E2^2+E3^2) 'Velocity Parameter
DEF FNPitchParam(E1,E2,E3)=E1/(SQR(E2^2+E3^2))   'Pitch Parameter
DEF FNYawParam(E2,E3)=E2/E3                'Yaw Parameter
DEF FNacos(X)
    IF X = 0 THEN
        acos = Pi/2
    ELSE
        acos = ATN(SQR(1-X^2)/X)
    END IF
    FNacos = acos
END DEF
DEF FNasin(X) = ATN(X/SQR(1-X^2))

SUB SelectThetas(THETA(1),Select1,Select2) STATIC
    Select1 = 1
    Select2 = 1
    DIM R(3)
    FOR I = 1 TO 3
        IF THETA(I) > Pi/2 THEN
            R(I) = ABS(Pi/2-(THETA(I)-Pi/2))/(THETA(I)-Pi/2)
        ELSE
            R(I) = ABS(Pi/2 - THETA(I))/THETA(I) 'Normalized
        END IF 'ratios indicating
        IF R(I) < 1 THEN R(I) = 1/R(I) 'distance from 0 or
        '90 degrees
    NEXT I
    IF R(1) > R(2) THEN Select1 = 2
    IF R(Select1) > R(3) THEN Select1 = 3
    R(Select1) = 1000
    'Select the two
    'angles farthest
    'from 0 or 90

```



```

        IF R(1) > R(2) THEN Select2 = 2
        IF R(Select2) > R(3) THEN Select2 = 3
END SUB

```

```

SUB ThetaLim(DC(2),THETA(1),S1,S2,S3,L,M,Z,LIM) STATIC
    DIM Qn(3,3)
    DIM FF(3)
    IJK = DC(S1,1)*DC(S2,1) + DC(S1,2)*DC(S2,2) + DC(S1,3)*DC(S2,3)
    L1 = DC(S1,1)*DC(S1,2)/M
    L2 = (L/M^2)*(DC(S1,2)^2+DC(S1,3)^2)
    LL = 2*(L1 - L2)
    B = (-4*Z*DC(S1,3)^2)
    B1B = (LL^2 - 4*Z*(DC(S1,2)^2+DC(S1,3)^2)/M^2)
    clim = SQR(B/B1B)*SIN(THETA(S1))
    lim = FNacos(clim)
    FOR J = 1 TO 3
        FF(J) = ((DC(S1,J)*COS(THETA(S1))) - DC(S2,J)*COS(lim))_
            /SIN(lim)
    NEXT J
    Tr = SIN(THETA(S1))/SIN(lim)
    N = (DC(S2,1)*FF(1) + DC(S2,2)*FF(2) + DC(S2,3)*FF(3))/Tr
    W1 = (N/M)*DC(S1,1)*DC(S1,2)
    W2 = (N*L/M^2)*(DC(S1,2)^2 + DC(S1,3)^2)
    W = 2*(W1 - W2)
    X = (N^2/M^2)*(DC(S1,2)^2 + DC(S1,3)^2) - DC(S1,3)^2
    W3 = SQR(abs(W^2 - 4*Z*X))
    Qn(S1,1) = -W/(2*Z)
    Qn(S1,2) = (N - Qn(S1,1)*L)/M
    Qn(S1,3) = (-Qn(S1,1)*DC(S1,1) - Qn(S1,2)*DC(S1,2))/DC(S1,3)
    if vect > 1.1 then stop
    SBETA = (-DC(S1,2)*COS(THETA(S1)) - Qn(S1,2)*SIN(THETA(S1)))
    BETA = FNasin(SBETA)
    SALPHA = DC(S1,1)*COS(THETA(S1))+Qn(S1,1)*SIN(THETA(S1))
    SALPHA = Pi/2 - (SALPHA/COS(BETA))
    ALPHA = FNasin(SALPHA)
    Slim = -DC(S3,1)*COS(ALPHA)*COS(BETA)_
        - DC(S3,2)*SIN(BETA)+DC(S3,3)*SIN(ALPHA)*COS(BETA)
    lim = FNacos(Slim)

```

```

END SUB

```

```

'*****
'*****
' *
' *
' *
' *
'*****
        WIDTH 80:CLS:KEY OFF
        Pi = 3.14159263

```

```

REDUCE:

```

```

        DIM DC(3,3)
        DIM Qn(3,3)
        DIM A(9):DIM V(3)
        DIM B(3,14) :DIM Qeff(3)
        DIM K(3,3):DIM THETA(3)
        DIM FF(3)
        OPEN "C:\PROBCAL\APARAM.DAT" FOR INPUT AS #1
OPEN "C:\PROBCAL\BPARAM.DAT" FOR INPUT AS #2
        FOR I = 0 TO 9 : INPUT#1,A(I):NEXT I
        FOR I = 0 TO 14
            FOR J = 1 TO 3
                INPUT#2,B(J,I)
            NEXT J
            PRINT
        NEXT I
        CLOSE #1,#2
        OPEN "C:\PROBCAL\KPARAM.DAT" FOR INPUT AS #2
        OPEN "C:\PROBCAL\DIRCOS" FOR INPUT AS #4           'get directional
        FOR I = 1 TO 3                                     'cosines
            FOR J = 1 TO 3
                INPUT#2,K(I,J)           'get velocity coeff
                INPUT#4,DC(J,I)
            NEXT J
        NEXT I
        CLOSE #4,#2
        INPUT "Enter data file name to reduce : ",FILE$
        INPUT "Enter data file name to store results : ",FILE1$
        OPEN FILE1$ FOR OUTPUT AS #2
        OPEN FILE$ FOR INPUT AS #1
        open "c:\probcad\theta.dat" for input as #5
        First = 1
BEGIN:
        WHILE NOT EOF(1)
SkipFirst:
            INPUT #1 ,V(1),V(2),V(3)
            IF First = 1 THEN                               'The first point is the
                First = 2                                    ' data point location so
                GOTO SkipFirst                               ' dont reduce it!
            END IF
        FOR I = 1 TO 3                                       'Calculate
            Qeff(I) = K(I,1)+K(I,2)*V(I)+K(I,3)*V(I)^2      'effective
                                                            'velocities
        NEXT I
            X= FNVelocityParam(V(1),V(2),V(3))             ' Velocity Parameter
            Y= FNPitchParam(V(1),V(2),V(3))                ' pitch parameters
            Z= FNYawParam(V(1),V(2))                       ' Yaw parameter
            Q=A(0)+A(1)*X+A(2)*X^2+A(3)*Y+A(4)*Y^2+A(5)*Z+A(6)*Z^2_
              +A(7)*X*Y+A(8)*X*Z+A(9)*Y*Z                  ' Velocity Magnitude
        FOR I = 1 TO 3                                       ' Caculate Thetas
            F=Qeff(I)/Q
            PARTA = B(I,0) + B(I,1)*Q + B(I,2)*Q^2 + B(I,3)*F_
                  + B(I,4)*F^2 + B(I,5)*Y + B(I,6)*Y^2 + B(I,7)*Z_

```



```

      + B(I,8)*Z^2
PARTB = B(I,9)*Q*F + B(I,10)*Q*Y + B(I,11)*Q*Z_
      + B(I,12)*F*Y + B(I,13)*F*Z + B(I,14)*Y*Z_
THETA(I) = PARTA + PARTB

```

```

NEXT I

```

```

CALL SelectThetas(THETA(),S1,S2)  'S1 & S2 will be the index
                                   'of the two angles farthest
                                   'from 0 or 90 degs

```

```

IF S1 > S2 THEN SWAP S1,S2
IF S1 = 1 AND S2 = 2 THEN S3 = 3
IF S1 = 2 AND S2 = 3 THEN S3 = 1
IF S1 = 1 AND S2 = 3 THEN S3 = 2
L=(DC(S2,3)*DC(S1,1)/DC(S1,3))-DC(S2,1)
M=(DC(S2,3)*DC(S1,2)/DC(S1,3))-DC(S2,2)
Z = (DC(S1,1) - DC(S1,2)*(L/M))^2 +(DC(S1,3)^2)*(1+(L^2/M^2))
CALL ThetaLim(DC(),THETA(),S1,S2,S3,L,M,Z,lim)
IF THETA(S3) > lim THEN SIGN = 1 ELSE SIGN = -1
FOR J = 1 TO 3
    FF(J)=(DC(S1,J)*COS(THETA(S1)) -
    DC(S2,J)*COS(THETA(S2)))/SIN(THETA(S2))
NEXT J
Tr = SIN(THETA(S1))/SIN(THETA(S2))
N=(DC(S2,1)*FF(1) + DC(S2,2)*FF(2) + DC(S2,3)*FF(3))/Tr
W1 = (N/M)*DC(S1,1)*DC(S1,2)
W2 = (N*L/M^2)*(DC(S1,2)^2 + DC(S1,3)^2)
W = 2*(W1 - W2)
X = (N^2/M^2)*(DC(S1,2)^2 + DC(S1,3)^2) - DC(S1,3)^2
Qn(S1,1) = (-W + (SIGN*SQR(ABS(W^2 - 4*Z*X))))/(2*Z)
Qn(S1,2) = (N - Qn(S1,1)*L)/M
Qn(S1,3) = (-Qn(S1,1)*DC(S1,1) - Qn(S1,2)*DC(S1,2))/DC(S1,3)
SBETA = (-DC(S1,2)*COS(THETA(S1)) - Qn(S1,2)*SIN(THETA(S1)))
BETA = FNasin(SBETA)
SALPHA = DC(S1,1)*COS(THETA(S1))+Qn(S1,1)*SIN(THETA(S1))
SALPHA = Pi/2 - (SALPHA/COS(BETA))
ALPHA = FNasin(SALPHA)
ALPHA = ALPHA*(180/Pi)
BETA = BETA*(180/Pi)
PRINT Q,ALPHA,BETA
PRINT#2, Q,ALPHA,BETA
input "hit a key";A$
WEND

```

*

```

QUIT:
END

```

REFERENCES

1. Bollinger, W.K., "Visualization of the Flow Around a Generic Destroyer Model in a Simulated Turbulent Atmospheric Boundary Layer", M.S. Thesis, Naval Postgraduate School, Monterey, California, September 1987.
2. Butler, T.L., Wagner, J.W., "Application of a Three-Sensor Hot-Wire Probe for Incompressible Flow", AIAA Journal, Vol 21 No. 5, pp 726-732, May 1983.
3. NASA Technical Memorandum 85871, Crossed Hot-Wire Data Acquisition and Reduction System, by R.V. Westphal and R.D. Mehta, January 1984.
4. DISA Information No 21, Medicinsk og Videnskabeligt Maleudstyr A/S, Mileparken 22, DK-2740 Skovlunde, Denmark, Measurement of Air Velocity by Means of a Triple Hot-Wire Probe, by C. Gaulier, pp 16-20, May 1983.
5. MetraByte Corporation, 440 Myles Standish Blvd., Taunton Mass. 02780, Dash-16/16F Manual, September 1986.
6. DISA Information No. 28, Medicinsk og Videnskabeligt Maleudstyr A/S, Mileparken 22, DK-2740 Skovlunde, Denmark, Effect of Combined Roll and Pitch Angles on Triple Hot-Wire Measurements of Mean and Turbulence Structure, by M.N. Frota and R.J. Moffat, pp 15-22, February 1983.

INITIAL DISTRIBUTION LIST

- | | | |
|----|--|---|
| 1. | Defense Technical Information Center
Cameron Station
Alexandria, VA 22304-615 | 2 |
| 2. | Library, Code 0142
Naval Postgraduate School
Monterey, CA 93943-5002 | 2 |
| 3. | Department Chairman, Code 67
Department of Aeronautics
Naval Postgraduate School
Monterey, CA 93943 | 1 |
| 4. | Commanding Officer
Naval Air Systems Command
Air Vehicle Division
Attn. Mr. John Ottensoser, Code Air 53011c
Jefferson Plaza 2, Rm 904 | 1 |
| 5. | Dr. J. Val Healey
Department of Aeronautics
Naval Postgraduate School
Monterey, CA 93943 | 4 |
| 6. | Lcdr. Gary J. Selman
101 Kalmia Forest Drive
Aiken SC 29803 | 2 |

Thesis
S41723 Selman
c.1 Calibration of a triple
wire probe for turbulence
measurements.

Thesis
S41723 Selman
c.1 Calibration of a triple
wire probe for turbulence
measurements.



thesS41723

Calibration of a triple wire probe for t



3 2768 000 78638 8

DUDLEY KNOX LIBRARY *CA*